PCT WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 7 : H04N 7/24, 5/00, 7/16 | A1

(11) International Publication Number: **WO 00/64180**

(43) International Publication Date: 26 October 2000 (26.10.00)

(21) International Application Number: PCT/US00/10017

(22) International Filing Date: 14 April 2000 (14.04.00)

(30) Priority Data:

60/130,328	21 April 1999 (21.04.99)	US
09/544,461	7 April 2000 (07.04.00)	US

(71) Applicant: GENERAL INSTRUMENT CORPORATION [US/US]; 101 Tournament Drive, Horsham, PA 19044 (US).

(72) Inventors: DEL SORDO, Christopher, S.; 229 Heatherfield Drive, Souderton, PA 18964 (US). GOFFIN, Glen, P., II; 351 Dublin Pike, Fountainville, PA 18923 (US). POLI, Christopher; 2607 Red Gate Drive, Doylestown, PA 18901 (US). MAKOFKA, Douglas, S.; 516 Fairhill Street, Willow Grove, PA 19090 (US). VINCE, Lawrence; 114 Aileen Drive, Lansdale, PA 19466 (US). BAHRAINI, Ardie; 411 Colonsay Court, Duluth, GA 30097 (US).

(74) Agent: KANANEN, Ronald, P.; Rader, Fishman & Grauer, Lion Building, Suite 501, 1233 20th Street, N.W., Washington, DC 20036 (US).

(81) Designated States: AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CU, CZ, DE, DK, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, UA, UG, UZ, VN, YU, ZW, ARIPO patent (GH, GM, KE, LS, MW, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

Published

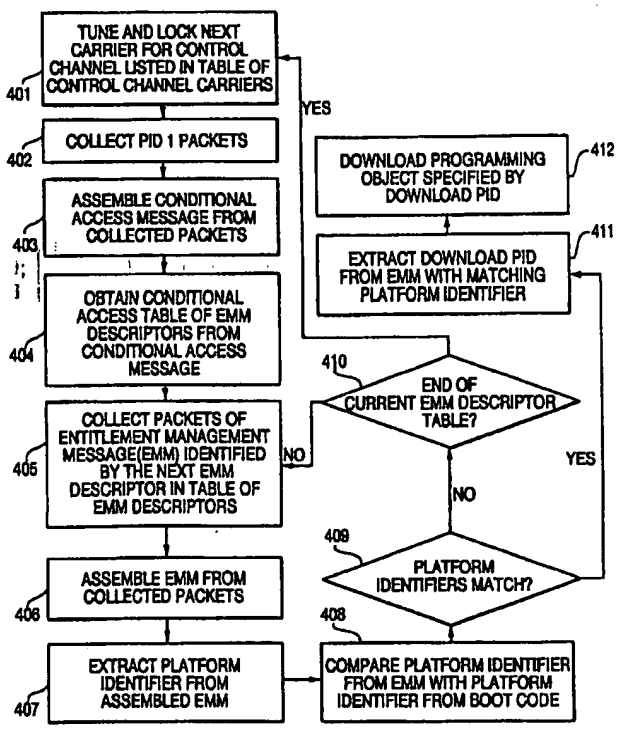
With international search report.

Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.

(54) Title: METHOD AND SYSTEM FOR IDENTIFYING AND DOWNLOADING APPROPRIATE SOFTWARE OR FIRMWARE SPECIFIC TO A PARTICULAR MODEL OF SET-TOP BOX IN A CABLE TELEVISION SYSTEM

(57) Abstract

A method and system identify programming code that is appropriate to the architecture and capabilities of a set-top terminal in a cable television system. The appropriate programming code is identified from among a variety of code objects being broadcast from the headend facility of the cable television system. A platform identifier stored in the set-top terminal is matched to a corresponding platform identifier in an entitlement management message or other download locator message that specifies where in the transport stream from the headend a particular code object can be acquired. By acquiring the object corresponding to the message bearing a matching platform identifier, the set-top terminal acquires programming code compatible with its attributes. Additionally, the cable television system can then optimally support a varied population of set-top terminals.



FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania	**ES**	Spain	**LS**	Lesotho	**SI**	Slovenia
AM	Armenia	**FI**	Finland	**LT**	Lithuania	**SK**	Slovakia
AT	Austria	**FR**	France	**LU**	Luxembourg	**SN**	Senegal
AU	Australia	**GA**	Gabon	**LV**	Latvia	**SZ**	Swaziland
AZ	Azerbaijan	**GB**	United Kingdom	**MC**	Monaco	**TD**	Chad
BA	Bosnia and Herzegovina	**GE**	Georgia	**MD**	Republic of Moldova	**TG**	Togo
BB	Barbados	**GH**	Ghana	**MG**	Madagascar	**TJ**	Tajikistan
BE	Belgium	**GN**	Guinea	**MK**	The former Yugoslav Republic of Macedonia	**TM**	Turkmenistan
BF	Burkina Faso	**GR**	Greece			**TR**	Turkey
BG	Bulgaria	**HU**	Hungary	**ML**	Mali	**TT**	Trinidad and Tobago
BJ	Benin	**IE**	Ireland	**MN**	Mongolia	**UA**	Ukraine
BR	Brazil	**IL**	Israel	**MR**	Mauritania	**UG**	Uganda
BY	Belarus	**IS**	Iceland	**MW**	Malawi	**US**	United States of America
CA	Canada	**IT**	Italy	**MX**	Mexico	**UZ**	Uzbekistan
CF	Central African Republic	**JP**	Japan	**NE**	Niger	**VN**	Viet Nam
CG	Congo	**KE**	Kenya	**NL**	Netherlands	**YU**	Yugoslavia
CH	Switzerland	**KG**	Kyrgyzstan	**NO**	Norway	**ZW**	Zimbabwe
CI	Côte d'Ivoire	**KP**	Democratic People's Republic of Korea	**NZ**	New Zealand		
CM	Cameroon			**PL**	Poland		
CN	China	**KR**	Republic of Korea	**PT**	Portugal		
CU	Cuba	**KZ**	Kazakstan	**RO**	Romania		
CZ	Czech Republic	**LC**	Saint Lucia	**RU**	Russian Federation		
DE	Germany	**LI**	Liechtenstein	**SD**	Sudan		
DK	Denmark	**LK**	Sri Lanka	**SE**	Sweden		
EE	Estonia	**LR**	Liberia	**SG**	Singapore		

TITLE OF THE INVENTION

Method and System for Identifying and Downloading Appropriate Software or Firmware Specific to a Particular Model of Set-Top Box in a Cable Television System

RELATED APPLICATIONS

This application claims priority from a previous U.S. provisional patent application entitled "Software and Firmware Initialization and Upgrade Management System and Method for an Advanced Set-Top Box in a Cable Television System," Serial No. 60/130,328, filed April 21, 1999.

FIELD OF THE INVENTION

The present invention relates to the field of initializing a set-top terminal of a cable television system and upgrading the software or firmware in the set-top terminal. More particularly, the present invention relates to the field of identifying and then downloading a specific version of a base platform code or other code object over the cable network that is appropriate to the architecture and capabilities of set-top terminal performing the download.

BACKGROUND OF THE INVENTION

In a typical cable television system, subscribers are provided with a set-top box or terminal. The set-top terminal is a box of electronic equipment that is used to connect the subscriber's television, and potentially other electronic equipment, with the cable network. The set-top box is usually connected to the cable network through a co-axial wall outlet.

The set-top box is essentially a computer that is programmed to process the signals from the cable network so as to provide the subscriber with the cable services. These services from the cable television company typically include access to a number of television channels and, perhaps, an electronic program guide. Additional premium channels may also be provided to subscribers at an additional fee. Pay-per-view events and video-on-demand may also be provided over the cable

network. The set-top box is programmed to provide these and other services to the subscriber.

However, the services of the cable company need not be limited to providing television programming. Some cable companies are now offering internet access and e-mail over their cable networks at speeds much faster than are available over conventional telephone lines. It is anticipated in the future that more and more services will be provided over the cable network, including even basic telephone service. Eventually, each home or office may have a single connection, via the cable network, to all electronic data services.

When a new set-top terminal is added to the cable network, it must be initialized. To initialize a set-top terminal, the terminal must be provided with the programming required to allow it to function within the specific cable network to which it is connected and to thereby provide the services for which the subscriber has paid. Additionally, as the cable network and the services provided evolve, the set-top terminal must also evolve to be able to provide subscribers with all the services of the cable network. This set-top box evolution will primarily involve changes to the programming, or perhaps a re-initialization, of the set-top box. By upgrading the soft- or firmware of the set-top box, the box can be made to perform more efficiently or offer new services as the cable network evolves.

In order to initialize new set-top terminals and upgrade the programming in the existing population of set-top boxes on a cable network, it is preferable to transmit the necessary programming to the set-top boxes via the cable network itself. Otherwise, a technician must visit each subscriber to install or upgrade the set-top boxes. Such field installations and upgrades would obviously be at significant expense. The headend is the facility from which the cable network operator broadcasts television signals and provides other services over the cable network. Software that is provided to the population of set-top terminals could be broadcast from the headend over the cable network.

However, there are a variety of problems associated with initializing and upgrading set-top terminals by broadcasting programming from the headend. For example, over time the population of set-top terminals will likely include different

makes and models of set-top terminals with different capacities. The software required to initialize or upgrade each make and model of set-top terminal may be different. Consequently, there is a need in the art for a method of matching the proper programming code to the capabilities of the set-top terminal being initialized or upgraded. Additionally, there is a need to automate the initialization process so as to eliminate or decrease the time required by a technician to install, upgrade or re-initialize a set-top terminal.

SUMMARY OF THE INVENTION

It is an object of the present invention to meet the above-described needs and others. Specifically, it is an object of the present invention to provide a method and mechanism for matching the proper programming code being broadcast over the cable plant to the capabilities of the set-top terminal being initialized or upgraded. Additionally, it is a further object of the present invention to automate the initialization process so as to eliminate or decrease the time required by a technician to install, upgrade or re-initialize a set-top terminal.

Additional objects, advantages and novel features of the invention will be set forth in the description which follows or may be learned by those skilled in the art through reading these materials or practicing the invention. The objects and advantages of the invention may be achieved through the means recited in the attached claims.

To achieve these stated and other objects, the present invention may be embodied and described as a method of identifying a code object for download by a set-top terminal from a data transport stream broadcast to the set-top terminal over a cable television system where the object identified is appropriate to the architecture and capabilities of the set-top terminal. The method is performed by matching a platform identifier stored in the set-top terminal with a platform identifier in a download locator message that specifies where in the data transport stream a particular code object can be acquired. The platform identifier is specific to the architecture and capabilities of the set-top terminal. Preferably, the download locator message is an entitlement management message.

Prior to comparing the platform identifiers, the method includes tuning the data transport stream with the set-top terminal based on a table of control channels carrying data transport streams. After tuning the data transport stream, the method proceeds by collecting PID 1 packets from the data transport stream and extracting from the data of those packets a table specifying packet identifiers for a group of download locator messages being transmitted on the data transport stream. With this table, the method proceeds by successively acquiring each of the download locator messages listed in the table and extracting from each download locator message a platform identifier. This continues until a download locator message is found bearing a platform identifier that matches the platform identifier stored in the set-top terminal.

After a match is found, the method proceed by obtaining locator data from the download locator message that has the platform identifier that matches the platform identifier stored in the set-top terminal. The locator data specifies where in the data transport stream a particular code object can be acquired. That particular code object will be appropriate for and compatible with the set-top terminal as indicated by the matched platform identifiers. The method then concludes with downloading to the set-top terminal the particular code object specified by the locator data from the download locator message that contains the platform identifier that matches the platform identifier stored in the set-top terminal.

The particular code object being acquired can be any of several different classes of objects. For example, the object can be a base platform code object, an operating system code object or a resident application code object.

The present invention also encompasses the necessary hardware to perform the method described above. For example, the present invention encompasses a system for of identifying a code object for download by a set-top terminal from a data transport stream broadcast to the set-top terminal over a cable television system where the object identified is appropriate to the architecture and capabilities of the set-top terminal. Such a system would minimally comprise means for obtaining a first platform identifier in a download locator message that specifies where in the data transport stream a particular code object can be acquired; and means for

matching the first platform identifier with a second platform identifier stored in the set-top terminal.

BRIEF DESCRIPTION OF THE DRAWINGS

The accompanying drawings illustrate the present invention and are a part of the specification. Together with the following description, the drawings demonstrate and explain the principles of the present invention.

Fig. 1 is a block diagram illustrating the three different stages at which different programming packages have control of the set-top terminal during the initialization process of the present invention.

Fig. 2 is a flow chart illustrating the steps of the initialization process for a set-top terminal according to the present invention.

Fig. 3 is a block diagram of the various memory devices and some code objects used in a set-top box according to the present invention.

Fig. 4 is a flow chart illustrated the method of the present invention for identifying code objects to be downloaded that are appropriate to the architecture and capabilities of the downloading set-top terminal.

DETAILED DESCRIPTION OF THE INVENTION:

The present invention addresses the problems involved in broadcasting a variety of programming over a cable television system for download by the population of set-top terminals connected to the network so as to initialize or upgrade those terminals where different programming objects being broadcast are appropriate to different specific classes of set-top terminals within the terminal population and each downloading terminal must identify and acquire the programming object or objects appropriate to its architecture and capabilities. This process includes providing those code objects to the set-top boxes that are necessary to allow those set-top boxes to function within the cable system or to upgrade the programming resident in different classes of set-top boxes so as to provide the services purchased by subscribers.

Stated in broad principle, the present invention aims to provide a set-top terminal architecture that includes a resident boot code object. As shown in Fig. 3, the boot code object (302) resides in the set-top terminal (300), preferably in read-only memory (ROM) (301) and can automatically execute and initialize or re-initialize the set-top terminal. The boot code will preferably be automatically executed by the central processor (321) of the set-top terminal. Execution of the boot code may be triggered by and immediately follow connection of power to the set-top terminal. The present invention may additionally require connection of the transport stream signal (322) from the cable system before execution of the boot code is triggered. Once the boot code is executing, no further action by the user/installer need be required. Moreover, no specific interaction is required between the headend and the set-top terminal that is initializing or booting.

As will be described in detail below, the boot code (302) of the present invention will automatically find, download and begin execution of the correct software code object or objects needed to initialize the set-top terminal. The boot code (302) will locate, identify and download the required programming from among potentially many code objects that might be multiplexed on the transport stream (322) coming from the headend facility of the cable television system. The boot code (302) recognizes the hardware configuration of the set-top terminal (300) in which it resides via an internal ROM coded identifier (320). This identifier (320) is matched against a value carried in an object download locator message from the transport stream (322) to insure that the boot code (302) obtains and downloads objects appropriate to the set-top terminal (300) in which the boot code (302) is resident.

Functionally, the boot code of the present invention will identify an appropriate control channel frequency, find the stream of control data packets within that control channel, identify and download the correct object from among the objects on the transport stream, verify that the downloaded code is authorized and error-free, and start the downloaded code without direct assistance by a technician or intervention from the headend. The term "boot code" as used herein comprises the minimal code needed to accomplish this functionality.

There are essentially two distinct phases of programming a set-top box addressed by the present invention. The first is the initial programming of the set-top box. The second is upgrading the programming or re-initialization of the set-top box after that box has been placed in service.

The initial programming of the set-top box is often performed by the cable system operator after the set-top box has been purchased from a manufacturer. Because each cable network is designed and built at different times by different service providers, each cable network may have a different design and architecture and use different code objects. Additionally, each system will likely have different classes of set-top terminals which were installed at different times and have different architectures and capabilities. Moreover, the specific services offered may vary among cable networks.

Therefore, to adapt the set-top boxes to function within the specific environment of a service provider's cable system and to provide the specific group of services currently offered by that particular service provider, each set-top box must be programmed accordingly or "initialized." Additionally, each terminal should, thereafter, be periodically re-programmed or upgraded to continue to function optimally within the evolving cable television system. Each time the programming of a set-top terminal is changed, the new code must be appropriate to the architecture and capabilities of that terminal.

The process of programming or reprogramming a set-top terminal according to the present invention will now be explained. In order for a set-top terminal to be initialized, i.e., accept and utilize the initial programming it receives, it must have some base programming that instructs it how to accept and use that initial programming. This base programming within the context of the present invention is called the boot code. As described above, the boot code is computer code resident in the permanent memory of the set-top terminal that is loaded, preferably into read-only memory, at the factory and cannot be changed once a terminal has been deployed.

As shown in Fig. 1, there are three general tiers or classifications of programming that run on or have control of the set-top terminal during different

stages in the initialization and operation of the terminal according to the present invention. Referring to Fig. 1, the first classification of code is the boot code (1). While running the boot code (1), the set-top terminal cannot provide any services to the subscriber. The function (2) of the boot code (1) is to search the data transport stream received from the headend facility to locate, acquire and begin execution of the base platform code (3) which is the next tier or classification of programming.

The boot code (1) is designed to authenticate the base platform code after the base platform code is downloaded. The boot code (1) will preferably re-authenticate the base platform code every time it launches the base platform object (3). When the base platform code (3) is executing, the execution of the boot code (1) is terminated and control of the set-top terminal passes to the base platform code (3).

The base platform code (3) may be factory loaded. However, under the principles of the present invention, the base platform code (3) is preferably transmitted to the set-top terminal from the cable headend during the initialization of the terminal. This allows the operator of the cable system to customize the base platform code (3) for optimal operation on the specific cable system where the set-top terminal is deployed. Preferably, the base platform code (3) is transmitted over the cable plant on an out-of-band (OOB) transport stream. However, it is within the scope of the present invention for the base platform code (3) to be transmitted on an in-band control channel.

The base platform code (3) has two functions. The first function of the base platform code (3) are to provide the basic capability of allowing a subscriber to watch television using the signal from the cable television system. The second function is to control the download (5) of the next classification of code objects, i.e., the target operating system (O/S) and resident applications (6). The base platform code (3), while allowing subscribers to watch television, does not generally support any additional functions of the set-top terminal. However, the base platform code (3) can acquire, authenticate, authorize and execute objects of the third and final classification of programming (e.g., the O/S) (5).

The third classification of programming, the operating system and resident applications (6) provide the additional set-top terminal functions available from the cable system. The operating system (O/S) is typically code from a third party (such as Microsoft's WinCE™) that provides access, with the resident applications, to all authorized set-top terminal capabilities. The operating system typically uses an additional embedded code module provided by the manufacturer of the set-top terminal which interfaces the operating system with the particular hardware of that set-top terminal to enable the operating system to function with that specific set-top terminal.

Resident applications are computer programs that run on the set-top terminal under the operating system. The resident applications work with the operating system to provide the capabilities of the set-top terminal that are in addition to watching television. The native suite is a specified group of software applications, including the operating system and perhaps various resident applications, that provide the intended functions of the set-top terminal. Specific elements of the native suite are determined by the system operator.

As indicated in Figs. 1 and 3, the boot code (1, 302) is preferably factory-loaded in the read-only memory (ROM) of the set-top terminal and is executed as soon as AC power is provided to the set-top terminal. Alternatively, the boot code may be executed in response to a reset signal (4) received, for example, from the headend, i.e., the system operator. This allows the system operator to re-initialize the set-top terminal whenever desired.

The reset signal (4) is preferably received by the base platform code (3) which then terminates execution of the operating system and resident applications (6), if running, and begins execution of the boot code (1). Alternatively, the reset signal (4) may cause the base platform code (3) to terminate and reload the native suite (6) rather than execute the boot code (1).

As described above, whenever executed, the boot code (1) acquires and loads the base platform code (2). The base platform code may be provided to the set-top terminal over the cable network from the headend or, alternatively, may be factory-loaded along with the boot code. The boot code (1) will either download the base

platform code (3), for example, over an out-of-band channel from the headend or, if the base platform code was factory-loaded, identify the base platform code (3) in memory. The boot code (1) authenticates the base platform code (3) from whatever source it is obtained and then executes the base platform code (3) .

The base platform code (3) then acquires the operating system and, preferably, the other objects of the native suite (6). The operating system and the other objects are downloaded from the headend over the cable network. The base platform code (3) will acquire the operating system and other objects when first executed or, while running, in response to an initialization message (4) from the system operator. The initialization message (4) maybe provided over the cable network. The operating system and resident applications (6) are then executed when the native suite is acquired, authorized and authenticated.

Fig. 2 is a flowchart providing a more detailed explanation of the initialization sequence according to the present invention. As shown in Fig. 2, when the set-top terminal is first powered, or an appropriate reset signal has been received, the boot code is executed (229). The boot code must first determine whether the set-top box has or must acquire the base platform code. To determine this, the boot code first checks the flash memory for the base platform code, the last known carrier (LKC) frequency of the control channel from the headend, and an Entitlement Management Message Provider Identification ("EMM Provider ID") (201, 202).

If any of three following conditions are discovered, the boot code will conclude that it must acquire the base platform code and will hunt for the out-of-band channel or the in-band channel from which the base platform code can be obtained. The boot code seeks to acquire the base platform code if (1) the base platform code, last known carrier and EMM Provider ID are not stored in the Flash memory, (2) the base platform code in the Flash memory fails an authentication check or (3) non-volatile memory indicates that hunting for the control channel (likely an out-of-band channel) is required.

If the Flash check determines that a base platform code object exists, the boot code proceeds to execute that base platform object after appropriate

authorization and authentication as described below. If both the base platform and the O/S are loaded in Flash, the boot code authorizes and authenticates the base platform and then launches the base platform and passes control of the set-top terminal thereto. The base platform object, in turn, authorizes and authenticates (A&A) the O/S. The authenticated O/S is then run and control passes to the O/S.

If the base platform code is not loaded in Flash memory, the boot code loads the base platform off of the out-of-band transport stream (203, 204, 207). However, before it is written to Flash memory, a successful authentication is required (206, 205). When the authenticated base platform code is executed, the boot code passes control to the base platform (211,228). If the base platform code fails the authentication check (205), the failed base platform code is deleted (208) and a counter is incremented (209) that tracks the number of attempts to acquire and authenticate a base platform code. If the counter is below a predetermined acceptable number of attempts, the base platform code is again downloaded (207). Alternatively, if the acceptable number of attempts to download the base platform code is exceeded, the set-top terminal may signal the headend for a service call (210).

Under the principles of the present invention, the boot code locates the base platform object using a boot code message or "bootcode_control_message" that is sent periodically on the out-of-band transport stream (204). Use of the bootcode_control_message will now be described in detail.

When the boot code determines the need to download the base platform object, it first hunts for the control channel. A table of possible carrier frequencies at which the control channel or channels are being broadcast is included in the boot code. These frequencies may be both in-band and out-of-band. The boot code will cause the set-top terminal to tune each of these frequencies in turn until the control channel is located and a carrier lock is obtained. If no control channel is received at a particular frequency for a predetermined period of time, the set-top terminal will tune the next frequency in the table.

The control channel is a stream of data packets that can be received and used by the set-top terminal. In order to broadcast a number of different objects

simultaneously, the headend will divide objects to be transmitted over the control channel into packets. The packets of the various objects being transmitted can then be interspersed or time-multiplexed together so that several objects are all transmitted essentially simultaneously. The packets for each particular object will have a common packet identifier or "PID." Thus, a set-top terminal can identify the packets for the object it is working to acquire. By acquiring all the packets with a particular PID, the complete object can then be reassembled by the set-top terminal from the set of packets with that particular PID.

According to the present invention, a set-top terminal can start anywhere in the progression to acquire an object and wrap around until all the necessary packets are downloaded. For example, the set-top terminal may load the first packet it receives with a PID X. That packet may be packet 50 of 100 marked by PID X. The terminal then continues to collect packets 51 to 100 with PID X, then 1 to 49. With all 100 packets obtained, the terminal can reassemble the packetized object.

Of particular concern to the present invention it the potential need to broadcast a number of objects simultaneously to accommodate different types or classes of set-top terminals in the population. Each class of set-top terminals may need a different version of, for example, the base platform code, the O/S or a resident application. Therefore, when the boot code is going to initialize the set-top terminal and must acquire the base platform code, the boot code must determine where to acquire the base platform appropriate to the set-top terminal on which it is running.

The process for identifying the correct object to download will now be described in detail with reference to Figs. 3 and 4. As shown in Fig. 3, the processor (321) of the set-top terminal (300) controls a tuner (323) to tune a control channel over which data and programming are being broadcast by the headend to the population of set-top terminals.

The set-top box (300) will have a table of carrier frequencies at which the headend may be broadcasting a control channel of data and programming. As shown in Fig. 4, the method of the present invention may begin with the set-top terminal tuning the first control channel listed in that table (401). Once the carrier

lock is achieved and the control channel is being received, the boot code will begin collecting packets from the transport stream on the control channel that are identified with PID 1 (402). PID 1 is dedicated to the conditional access message in the MPEG standard. The packets of PID 1 will provide the boot code running on the set-top terminal with a Conditional Access Table (CAT) of EMM Provider IDs each of which identifies a PID for a set of packets on the transport stream that constitute an EMM stream (Entitlement Management Message) (403, 404).

The boot code will begin with the first EMM Provider ID and begin loading packets from the transport stream that are marked with the EMM PID given by the first EMM Provider ID (405). The EMM PID packets being acquired will contain a boot code message of the present invention which, in turn, includes a platform identifier. Thus, the Entitlement Management Message will be extracted from the EMM PID packets acquired (406) and the platform identifier from the EMM will be extracted (407).

As shown in Fig. 3, the boot code (302) which is factory-installed in the set-top terminal will also include a platform identifier (320) that is specific to the type, architecture and capabilities of terminal (300) in which the boot code is resident. When running, the boot code will attempt to match the platform identifier provided at the factory with the platform identifier from the boot code message of the EMM PID packets (408, 409).

If no match is found, the boot code will select the next EMM Provider ID in the CAT and check the packets of the EMM PID identified by that EMM Provider ID for a boot code message with a matching platform identifier (410, 405). This continues until the matching platform identifier is found . It may be possible to search multiple EMM PID's simultaneously to reduce the EMM validation time and the time required to find the matching boot code message.

If all the EMM Provider IDs in the CAT of PID 1 are checked and no match is found for the platform identifier (410), the boot code will look for another control channel on another carrier frequency by returning to the table of carrier frequencies (401). When another frequency with a control channel is identified and locked, the boot code will extract PID 1 and repeat the process outlined above. This continues

until a boot code message with a platform identifier matching the platform identifier of the boot code is found.

When the boot code finds a boot code message with a matching platform identifier, the boot code will extract a download PID (DL PID) specified by the EMM with the matching platform identifier (411). The download PID (DL PID) is the identifier for the packets that carry the code object, e.g., the base platform code object, that is appropriate for the type of set-top terminal (300) with the platform identifier (320). The boot code can then download the base platform code or other code object by acquiring the packets with the DL PID and reassembling the data in those packets into the base platform code.

As will be understood by those in the art, the platform identifier (320) of the present invention can be used to check any type of code object for is compatibility with the terminal in which the identifier (320) is resident. The invention is not limited to the use of the identifier (320) by the boot code (302) to locate and identify an appropriate base platform code. The platform identifier (320) of the present invention can be used in the same manner described above, for example, by the base platform code to identify and acquire an operating system object designed for the downloading set-top terminal. The platform identifier (320) can also be used to identify other elements of the native suite, i.e., resident applications, that are appropriate for the downloading set-top terminal.

Additionally, the platform identifier (320) of the present invention need not necessarily be incorporated into the boot code (302). Rather, the platform identifier (320) can be stored anywhere within the set-top terminal (300) where it can be accessed by the executing programs that require it to identify appropriate code objects for download.

As shown in Fig. 2, once the base platform code has been downloaded or identified as already resident in Flash memory, an authentication check (206) is performed to verify that the base platform code has been accurately and completely received and has not been altered by an unauthorized party. If the base platform fails the authentication check, it is deleted (208). A load counter may then be checked to determine the number of times the set-top terminal has attempted to

acquire a valid base platform code (209). If the counter exceeds a predetermined limit, the set-top terminal may signal the headend for a service call or may indicate the need to request a service call to the subscriber (210). If the load counter is not exceeded, the boot code will revert to the process described above and attempt again to download the base platform code (207).

Alternatively, if the base platform code is authenticated, it is then launched (211). The base platform code will then determine if the native suite, including the O/S, is loaded in the Flash memory (214). If it is not, the base platform code will seek to download the native suite.

With the base platform code running, the system operator may provide the set-top terminal with a set of "initialization messages" that provide, for example, channel maps, tables and EMM information (219, 212). These messages should be provided before the native suite is loaded. The initialization messages may instruct the set-top terminal where to acquire the native suite.

After the native suite has been downloaded, or is found already existing in Flash memory, an authorization check is performed on the native suite (215, 220, 224, 223). The download of the native suite will include an Object Conditional Access Message (OCAM) that is recorded by the set-top terminal. The authentication signature and authorization code for the native suite object are provided in the OCAM and used to authorize and authenticate the native suite in the manner described below.

If the authorization check is not successful, the native suite code will be deleted (225, 217) and the base platform code will again attempt to acquire the native suite (221). If the authorization check is successful, the native suite and any resident applications associated with it, are loaded and an authentication check is performed (222). As before, if the authentication check fails, the downloaded code will be deleted (217) and a load counter will be checked (216) to see if another attempt to download the code should be made or a service call signaled (213).

Alternatively, if the authentication check (222, 218) is successful, the native suite and any associated resident applications will be executed beginning with the O/S (226, 227). The base platform code performs the authorization and

authentication on the O/S code. If the O/S passes the authorization and authentication checks, the O/S is executed and control is transferred to the O/S. The BIOS (Basic Input/Output Software) may perform the authorization and authentication of the remainder of the native suite (215, 224, 222).

In summary, various portions of the boot process include an object authorization and authentication (A&A) process for newly acquired or located objects. The authorization check of the native suite is done within the base platform. The authorization of the base platform is, in turn performed by the boot code, which can only authenticate a base platform object. When running, the O/S of the native suite performs the authentication and authorization of subsequently loaded objects. These checks are required so that, given an interruption in power, etc., the authorization status of the terminal can be verified. If, at any point an authorization or authentication check fails, the object being checked is disabled.

Authentication is performed as follows. When a code object is broadcast over the cable network, it is associated with an authorization code and an authentication signature. For the base platform object, the authorization code is preferably given in an object_id field of the boot code message. The authentication signature is preferably given in an object_description field of the boot code message. For other objects, such as the O/S and the native suite, the authorization code and authentication signature are provided in an OCAM downloaded with the object.

The authentication signature is computed mathematically using a specific algorithm with the code object itself as the input for the algorithm. The signature is then re-computed by the set-top terminal using the same algorithm and the downloaded code as input. If the signature computed by the set-top terminal matches the one transmitted with the code, the code can be implemented with confidence that its has been transmitted properly, without inadvertent or malicious alteration.

The present invention provides for two basic ways to upgrade the basic platform in a population of set-top terminals once those terminals have been placed in full service. These two methods of upgrade are (1) a universal upgrade of the

entire population (i.e., the entire population tuned to a particular control stream) and (2) a targeted upgrade of a subset or subsets of the population. Both methods may make use of the boot code to perform the upgrade.

A universal upgrade is accomplished by broadcasting an order from the headend for all set-top terminals on the control stream to delete their existing base platform object. The boot code then begins executing, assumes control, and performs the initialization procedure outlined above, including replacing the deleted base platform with a base platform downloaded over the cable network.

A targeted upgrade applies to a single terminal or a small group of terminals on a given control channel. Each terminal has a specific single-cast address and can, therefore, be addressed by the headend and instructed to delete the existing base platform code and re-initialize with upgrade code. Alternatively, each terminal has one or more multi-cast addresses that are shared by other terminals in the population. Four such multi-cast addresses for each terminal are preferred. With a multi-cast address, the headend can signal a code purge and re-initialization for a specific class of terminals that share that particular multi-cast address.

In a targeted upgrade, the base platform, using standard download messages, sets up download parameters in a start-up database in non-volatile memory (See Fig. 3) and allows the boot code to take control. The boot code then uses the parameters to acquire the upgraded base platform code, replacing the original base platform code. This is done while the older version of the base platform code is still spinning at a location indicated by the boot message.

In addition to the examples given above, an upgrade need not disturb the base platform code. Rather, the upgrade or reset signal, whether universal or targeted, may instruct the set-top terminal(s) to terminate and delete only the operating system (O/S), the entire native suite, or one or more particular resident applications. Control then returns to the base platform code which will acquire and authenticate a new O/S, entire native suite, or portions of the native suite as necessary. In this way, the native suite (or just the O/S) can be upgraded without requiring the base platform code to be reacquired as well.

Fig. 3 illustrates four memory units of a set-top terminal (300) according to the present invention. A read-only memory unit (ROM) (301) contains the boot code (302). A flash memory unit (303) contains the base platform code (304) and the O/S object (306). Aside from these objects, additional flash memory is available (305). Two stack pointers (307, 308) designate absolute locations in the Flash memory (303) for the base platform code (304, 308) and the O/S (306, 307). It is important that these two objects are always located at the same location in Flash (303).

A non-volatile memory unit (310) preferably has both a managed and a non-managed segment. The base platform code (304) may store parameters and other data in the non-managed portion of the non-volatile memory unit (310).

Finally, a random access memory unit (RAM) (309) is provided. Downloaded objects such as the base platform code, the O/S, etc. may be stored in the RAM (309) until authenticated. Once authorization and authentication are successfully completed, the objects may be transferred from the RAM (309) to the Flash memory unit (303) for long-term storage.

The preceding description has been presented only to illustrate and describe the invention. It is not intended to be exhaustive or to limit the invention to any precise form disclosed. Many modifications and variations are possible in light of the above teaching.

The preferred embodiment was chosen and described in order to best explain the principles of the invention and its practical application. The preceding description is intended to enable others skilled in the art to best utilize the invention in various embodiments and with various modifications as are suited to the particular use contemplated.

WHAT IS CLAIMED IS:

1. A method of identifying a code object for download by a set-top terminal from a data transport stream broadcast to the set-top terminal over a cable television system where the object identified is appropriate to the architecture and capabilities of the set-top terminal, the method comprising matching a platform identifier stored in said set-top terminal with a platform identifier in a download locator message that specifies where in said data transport stream a particular code object can be acquired, wherein said platform identifier is specific to said architecture and capabilities of said set-top terminal.

2. The method of claim 1, wherein said download locator message is an entitlement management message.

3. The method of claim 1, further comprising tuning said data transport stream with said set-top terminal based on a table of control channels carrying data transport streams.

4. The method of claim 1, further comprising collecting PID 1 packets from said data transport stream and extracting therefrom a table specifying packet identifiers for a plurality of download locator messages being transmitted on said data transport stream.

5. The method of claim 4, further comprising successively acquiring said plurality of download locator messages according to said table and extracting platform identifiers therefrom until a download locator message is found bearing a platform identifier that matches said platform identifier stored in said set-top terminal.

6. The method of claim 5, further comprising obtaining locator data from said download locator message having a platform identifier that matches said

platform identifier stored in said set-top terminal, wherein said locator data specifies where in said data transport stream a particular code object can be acquired.

7. The method of claim 6, further comprising downloading to said set-top terminal said particular code object specified by said locator data from said download locator message having a platform identifier that matches said platform identifier stored in said set-top terminal.

8. The method of claim 1, wherein said code object is a base platform code object.

9. The method of claim 1, wherein said code object is an operating system code object.

10. The method of claim 1, wherein said code object is a resident application code object.

11. A system for of identifying a code object for download by a set-top terminal from a data transport stream broadcast to the set-top terminal over a cable television system where the object identified is appropriate to the architecture and capabilities of the set-top terminal, the system comprising:

means for obtaining a first platform identifier in a download locator message that specifies where in said data transport stream a particular code object can be acquired; and

means for matching said first platform identifier with a second platform identifier stored in said set-top terminal,

wherein said platform identifier is specific to said architecture and capabilities of said set-top terminal.

12. The system of claim 11, wherein said download locator message is an entitlement management message.

13. The system of claim 11, further comprising means for tuning said data transport stream with said set-top terminal based on a table of control channels carrying data transport streams.

14. The system of claim 11, further comprising:

means for collecting PID 1 packets from said data transport stream; and

means for extracting therefrom a table specifying packet identifiers for a plurality of download locator messages being transmitted on said data transport stream.

15. The system of claim 14, further comprising means for successively acquiring said plurality of download locator messages according to said table and extracting platform identifiers therefrom until a download locator message is found bearing a first platform identifier that matches said second platform identifier stored in said set-top terminal.

16. The system of claim 15, further comprising means for obtaining locator data from said download locator message having said first platform identifier that matches said second platform identifier stored in said set-top terminal, wherein said locator data specifies where in said data transport stream a particular code object can be acquired.

17. The system of claim 16, further comprising means for downloading to said set-top terminal said particular code object specified by said locator data from said download locator message having said first platform identifier that matches said second platform identifier stored in said set-top terminal.

18. The system of claim 11, wherein said code object is a base platform code object.

19. The system of claim 11, wherein said code object is an operating system code object.

20. The system of claim 11, wherein said code object is a resident application code object.

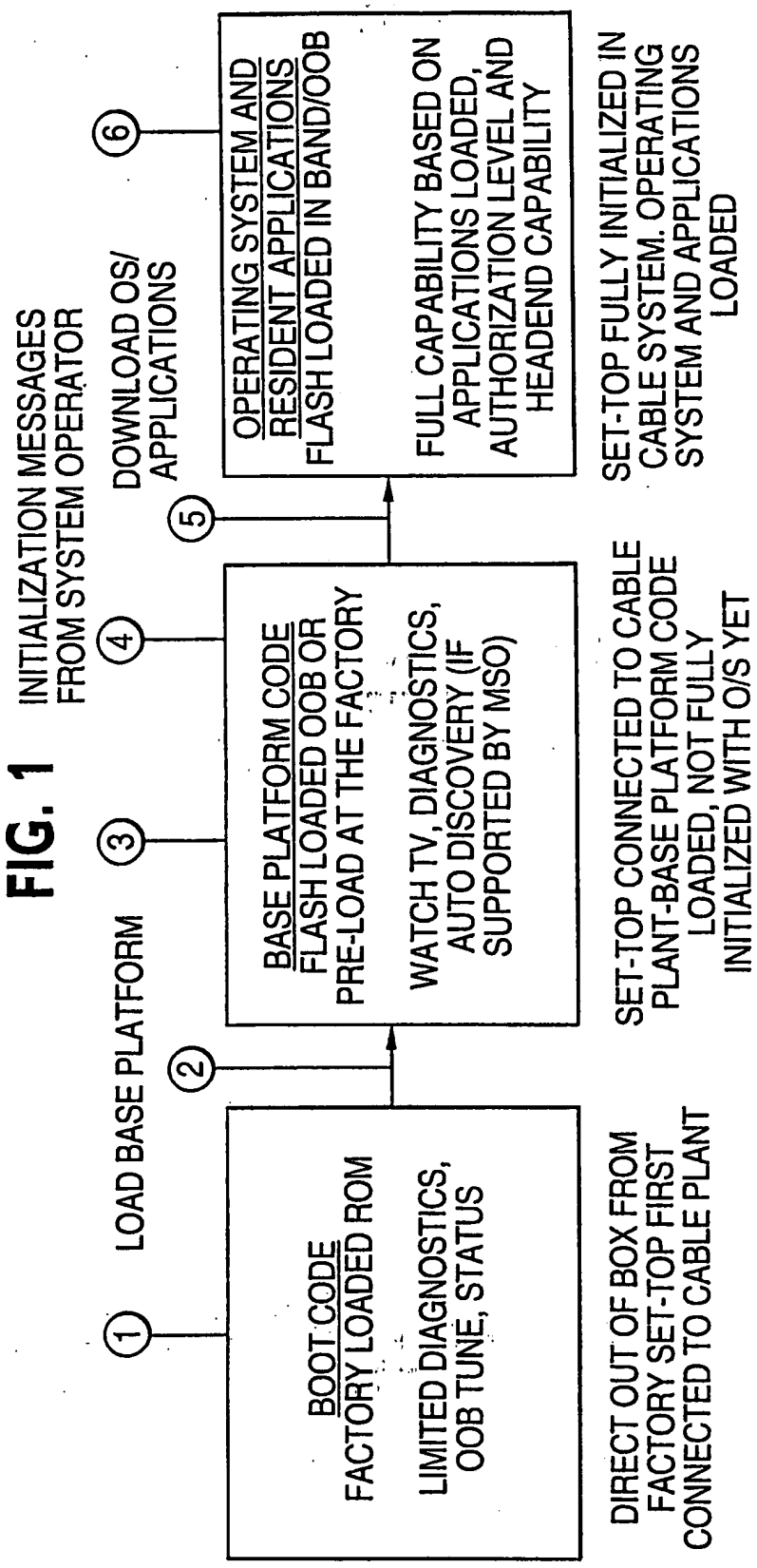


FIG. 1
INITIALIZATION MESSAGES FROM SYSTEM OPERATOR
1
LOAD BASE PLATFORM
2
3
4
DOWNLOAD OS/ APPLICATIONS
5
6
BOOT CODE
FACTORY LOADED ROM
LIMITED DIAGNOSTICS, OOB TUNE, STATUS
DIRECT OUT OF BOX FROM FACTORY SET-TOP FIRST CONNECTED TO CABLE PLANT
BASE PLATFORM CODE
FLASH LOADED OOB OR PRE-LOAD AT THE FACTORY
WATCH TV, DIAGNOSTICS, AUTO DISCOVERY (IF SUPPORTED BY MSO)
SET-TOP CONNECTED TO CABLE PLANT-BASE PLATFORM CODE LOADED, NOT FULLY INITIALIZED WITH O/S YET
OPERATING SYSTEM AND RESIDENT APPLICATIONS
FLASH LOADED IN BAND/OOB
FULL CAPABILITY BASED ON APPLICATIONS LOADED, AUTHORIZATION LEVEL AND HEADEND CAPABILITY
SET-TOP FULLY INITIALIZED IN CABLE SYSTEM. OPERATING SYSTEM AND APPLICATIONS LOADED

FIG. 2

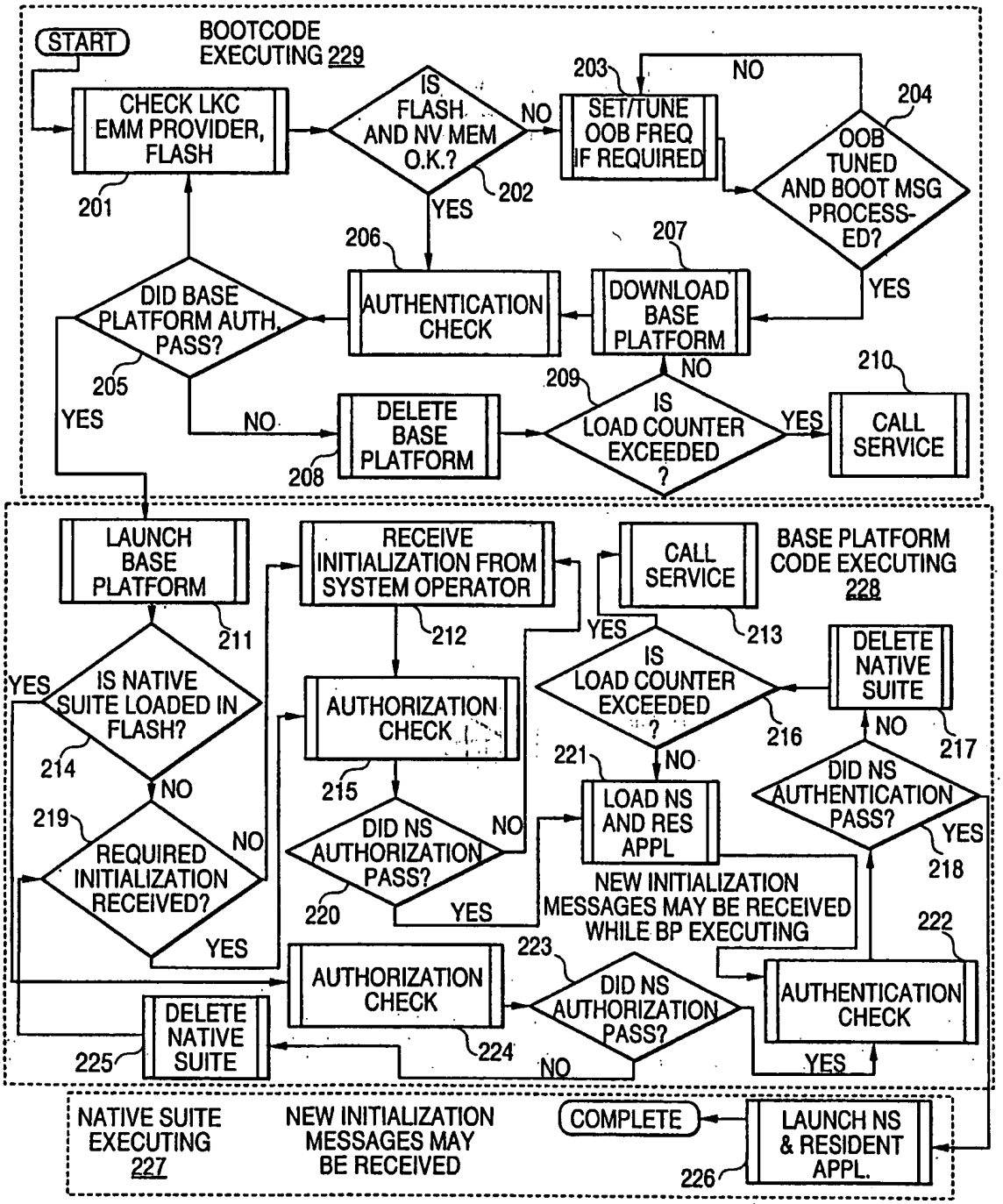

START
BOOTCODE EXECUTING 229
CHECK LKC EMM PROVIDER, FLASH
201
IS FLASH AND NV MEM O.K.?
202
NO
YES
SET/TUNE OOB FREQ IF REQUIRED
203
NO
OOB TUNED AND BOOT MSG PROCESS-ED?
204
YES
DOWNLOAD BASE PLATFORM
207
AUTHENTICATION CHECK
206
DID BASE PLATFORM AUTH. PASS?
205
YES
NO
DELETE BASE PLATFORM
208
IS LOAD COUNTER EXCEEDED ?
209
NO
YES
CALL SERVICE
210
LAUNCH BASE PLATFORM
211
RECEIVE INITIALIZATION FROM SYSTEM OPERATOR
212
CALL SERVICE
213
BASE PLATFORM CODE EXECUTING 228
IS NATIVE SUITE LOADED IN FLASH?
214
YES
NO
AUTHORIZATION CHECK
215
IS LOAD COUNTER EXCEEDED ?
216
YES
NO
DELETE NATIVE SUITE
217
DID NS AUTHENTICATION PASS?
218
YES
NO
REQUIRED INITIALIZATION RECEIVED?
219
NO
YES
DID NS AUTHORIZATION PASS?
220
NO
YES
LOAD NS AND RES APPL
221
NEW INITIALIZATION MESSAGES MAY BE RECEIVED WHILE BP EXECUTING
AUTHORIZATION CHECK
224
DID NS AUTHORIZATION PASS?
223
NO
YES
AUTHENTICATION CHECK
222
DELETE NATIVE SUITE
225
NATIVE SUITE EXECUTING 227
NEW INITIALIZATION MESSAGES MAY BE RECEIVED
COMPLETE
LAUNCH NS & RESIDENT APPL.
226

FIG. 3

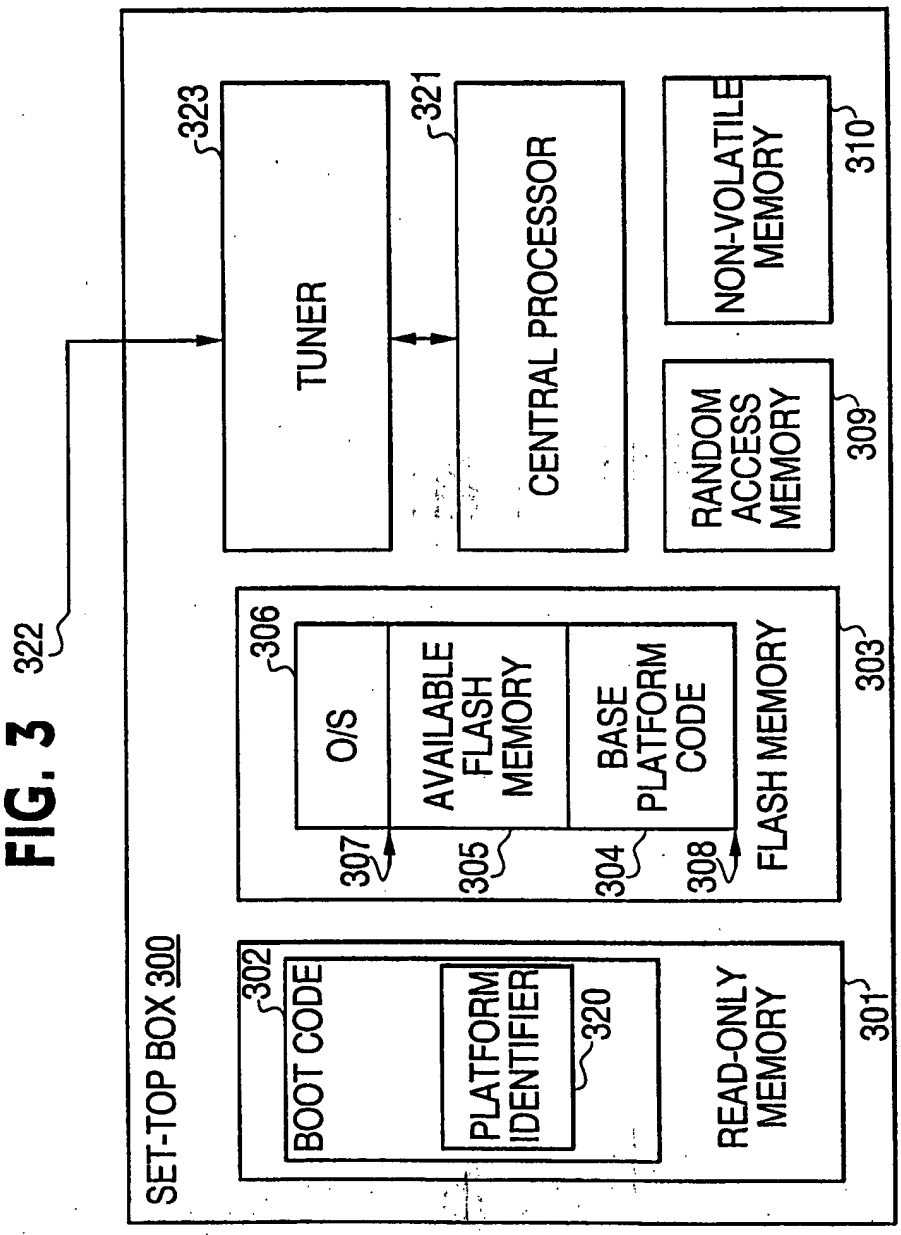
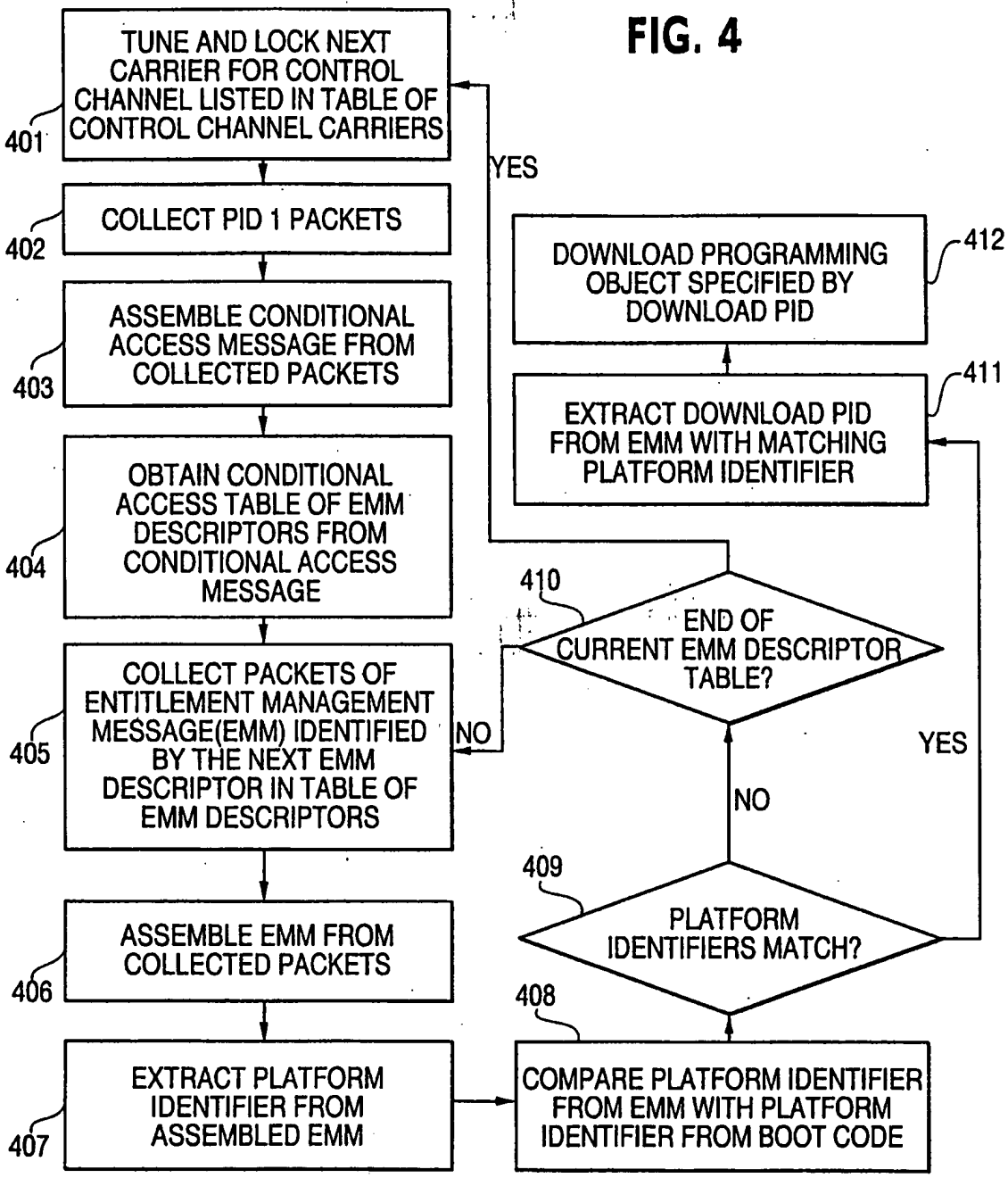

322
SET-TOP BOX 300
323
TUNER
321
CENTRAL PROCESSOR
NON-VOLATILE MEMORY
310
RANDOM ACCESS MEMORY
309
306
O/S
AVAILABLE FLASH MEMORY
BASE PLATFORM CODE
FLASH MEMORY
303
307
305
304
308
302
BOOT CODE
PLATFORM IDENTIFIER
320
READ-ONLY MEMORY
301

FIG. 4


TUNE AND LOCK NEXT CARRIER FOR CONTROL CHANNEL LISTED IN TABLE OF CONTROL CHANNEL CARRIERS
401
COLLECT PID 1 PACKETS
402
ASSEMBLE CONDITIONAL ACCESS MESSAGE FROM COLLECTED PACKETS
403
OBTAIN CONDITIONAL ACCESS TABLE OF EMM DESCRIPTORS FROM CONDITIONAL ACCESS MESSAGE
404
COLLECT PACKETS OF ENTITLEMENT MANAGEMENT MESSAGE(EMM) IDENTIFIED BY THE NEXT EMM DESCRIPTOR IN TABLE OF EMM DESCRIPTORS
405
ASSEMBLE EMM FROM COLLECTED PACKETS
406
EXTRACT PLATFORM IDENTIFIER FROM ASSEMBLED EMM
407
COMPARE PLATFORM IDENTIFIER FROM EMM WITH PLATFORM IDENTIFIER FROM BOOT CODE
408
PLATFORM IDENTIFIERS MATCH?
409
END OF CURRENT EMM DESCRIPTOR TABLE?
410
EXTRACT DOWNLOAD PID FROM EMM WITH MATCHING PLATFORM IDENTIFIER
411
DOWNLOAD PROGRAMMING OBJECT SPECIFIED BY DOWNLOAD PID
412
YES
NO
YES
NO

INTERNATIONAL SEARCH REPORT

Int: ional Application No
PCT/US 00/10017

A. CLASSIFICATION OF SUBJECT MATTER
IPC 7 H04N7/24 H04N5/00 H04N7/16

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 7 H04N G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)
EPO-Internal, WPI Data, PAJ, INSPEC

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	WO 97 30549 A (POWERTV INC) 21 August 1997 (1997-08-21)	1,3, 8-11,13, 18-20
Y	the whole document	2,4-7, 12,14-17
Y	US 5 420 866 A (WASILEWSKI ANTHONY J) 30 May 1995 (1995-05-30) section "Summary of the invention" column 11, line 21 -column 13, line 34 column 14, line 45 -column 15, line 17 column 16, line 17 - line 64 column 18, line 44 -column 19, line 5	2,12
A	abstract; figures 3A,3B,5,6,7B,8B	1,4-7, 11,14-17

-/--

[X] Further documents are listed in the continuation of box C.

[X] Patent family members are listed in annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
21 August 2000	29/08/2000

Name and mailing address of the ISA	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL – 2280 HV Rijswijk Tel. (+31–70) 340–2040, Tx. 31 651 epo nl, Fax: (+31–70) 340–3016	La, V

1

Form PCT/ISA/210 (second sheet) (July 1992)

INTERNATIONAL SEARCH REPORT

Int. Jonal Application No
PCT/US 00/10017

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication,where appropriate, of the relevant passages	Relevant to claim No.
X Y A	EP 0 907 285 A (CANAL PLUS SA) 7 April 1999 (1999-04-07) the whole document	1,11 4-7, 14-17 3,8-10, 13,18-20
X A	US 5 666 293 A (HUDSON JR HENRY G ET AL) 9 September 1997 (1997-09-09) column 4, line 7 -column 6, line 10 column 6, line 50 -column 39, line 37 abstract; figure 1	1,9-11, 19,20 2-8, 12-18
A	US 5 734 589 A (HUDSON JR HENRY G ET AL) 31 March 1998 (1998-03-31) column 1, line 38 -column 2, line 24 column 3, line 60 -column 6, line 6 column 7, line 54 - line 60 column 11, line 43 -column 16, line 30 column 21, line 39 - line 61 column 26, line 56 -column 28, line 25 abstract; figure 5	1-20
A	US 5 440 632 A (BACON KINNEY C ET AL) 8 August 1995 (1995-08-08) column 2, line 3 -column 3, line 64 column 5, line 59 - line 64 column 9, line 25 -column 11, line 20 column 13, line 54 -column 15, line 68 abstract; figure 1	1-20
A	EP 0 905 984 A (MATSUSHITA ELECTRIC IND CO LTD) 31 March 1999 (1999-03-31) section '005! abstract	1-20
A	US 5 852 290 A (CHANEY JOHN WILLIAM) 22 December 1998 (1998-12-22) column 3, line 57 - line 66 column 11, line 23 - line 34	1,2,4-7, 11,12, 14-17

1

INTERNATIONAL SEARCH REPORT

Information on patent family members

Int ional Application No
PCT/US 00/10017

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
WO 9730549 A	21-08-1997	US 5951639 A	14-09-1999
		AU 1693597 A	02-09-1997
		EP 0880857 A	02-12-1998
US 5420866 A	30-05-1995	AU 687844 B	05-03-1998
		AU 7220994 A	17-10-1995
		CA 2186368 A,C	05-10-1995
		JP 2940639 B	25-08-1999
		JP 9511369 T	11-11-1997
		WO 9526597 A	05-10-1995
EP 0907285 A	07-04-1999	AU 9278398 A	27-04-1999
		EP 1020078 A	19-07-2000
		WO 9918724 A	15-04-1999
		NO 20001651 A	05-06-2000
		ZA 9808996 A	24-05-1999
US 5666293 A	09-09-1997	US 5734589 A	31-03-1998
		US 5635979 A	03-06-1997
		US 5768539 A	16-06-1998
		US 5978855 A	02-11-1999
		AU 2657995 A	21-12-1995
		WO 9533338 A	07-12-1995
US 5734589 A	31-03-1998	US 5666293 A	09-09-1997
		US 5768539 A	16-06-1998
		US 5978855 A	02-11-1999
US 5440632 A	08-08-1995	US 5715515 A	03-02-1998
EP 0905984 A	31-03-1999	JP 11098477 A	09-04-1999
		AU 716549 B	02-03-2000
		AU 8703498 A	15-04-1999
US 5852290 A	22-12-1998	AU 701593 B	04-02-1999
		AU 3239495 A	14-03-1996
		BR 9508621 A	30-09-1997
		EP 0782807 A	09-07-1997
		FI 970677 A	18-02-1997
		JP 10506507 T	23-06-1998
		PL 318647 A	07-07-1997